UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/449,021	11/24/1999	HELMUT EMMELMANN	EMME-1000US0	5718

8685 7590 02/24/2012

DERGOSITS & NOAH LLP
Three Embarcadero Center
Suite 410
SAN FRANCISCO, CA 94111

EXAMINER	
KENDALL, CHUCK O	

ART UNIT	PAPER NUMBER
2192	

MAIL DATE	DELIVERY MODE
02/24/2012	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE BOARD OF PATENT APPEALS
AND INTERFERENCES

Ex parte HELMUT EMMELMANN

Appeal 2012-001997
Application 09/449,021
Technology Center 2100

Before MARC S. HOFF, CARLA M. KRIVAK, and THOMAS S. HAHN, *Administrative Patent Judges*.

HOFF, *Administrative Patent Judge*.

DECISION ON APPEAL

STATEMENT OF CASE

Appellant appeals under 35 U.S.C. § 134 from a Final Rejection of claims 1, 2, 22, 23, 26, 30, 32, 33, 41, 42, 59-63, 67-69, and 71-73.[1,2] We have jurisdiction under 35 U.S.C. § 6(b).

We reverse.

Appellant's invention concerns developing server-side web applications by composing them out of interactive server-side components and placing them on a page template. The invention includes a browser-based component editor that allows a user to remotely edit a running application. A server transforms each page into a new page that contains editing features (such as scripts and handles) for the page itself (Spec. 4-5).

Claims 1 and 26 are exemplary of the claims on appeal:

1. A computer-readable medium encoded with computer programs having executable instructions for editing software applications that run on a data network which couples a server computer and a client computer, wherein the client computer runs a browser program, and whereupon request by the browser program, at least one of the applications generates generated documents for display by the browser program on a display device and responds to the request with the generated documents, comprising:

a document generator program running at least part of one of the applications being edited and generating the generated documents, said generated documents including additional editing features for interpretation by the browser program; and

---

[1] Claims 6, 8, 51-58, 74-96, and 114-128 stand allowable. Claims 3-5, 24, 25, 27-29, 31, 43, 64-66, and 70 stand objected to as dependent upon a rejected base claim, but would be allowable if rewritten in independent form. Claims 9-21, 34-40, 44-50, and 97-113 stand withdrawn as directed to non-elected subject matter. Claim 7 has been cancelled.

[2] We note the petition decision mailed November 22, 2011, denying Appellant's request to reinstate claims 3-5, 24, 25, 27-29, 31, 43, 64-66, and 70.

an editor program dynamically operating on the generated documents displayed by the browser program via the editing features.

26. A system having a data network which couples a server computer to a client computer, the server computer running an application to modify dynamic documents on the server computer, the server computer comprising:

a document store;

a first software program including instructions for transforming at least one first document retrieved from the document store into a second document having features which permit editing of the first document such that at least a part of the second document appears and functions similar to the run-time view of the first document; and

a second software program including instructions to receive information from the client computer and instructions to modify the first document stored in the document store.

The Examiner relies upon the following prior art in rejecting the claims on appeal:

Faustini US 5,842,020 Nov. 24, 1998

Arturo Crespo & Eric A. Bier, *WebWriter: A Browser-Based Editor for Constructing Web Applications*, 28 COMPUTER NETWORKS & ISDN SYS. 1291 (1996) [hereinafter WebWriter].

Arturo Crespo et al., *Responsive Interaction for a Large Web Application: The Meteor Shower Architecture in the WebWriter II Editor*, 29 COMPUTER NETWORKS & ISDN SYS. 1501 (1997) [hereinafter WebWriter II]

Claims 1, 2, 22, 23, 41, and 42 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over WebWriter II in view of Faustini.

Claims 26, 30, 32, 33, 59-63, 67-69, and 71-73 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over WebWriter in view of Faustini.

Throughout this decision, we make reference to the Appeal Brief ("App. Br.," filed June 9, 2010), the Reply Brief ("Reply Br.," filed Sep. 12,

2011), and the Examiner's Answer ("Ans.," mailed July 12, 2011) for their respective details.

ISSUES

Appellant argues generally, with respect to all claims, that none of the cited references teach or suggest editing a "dynamic" web application that is running at the server (Reply Br. 4).

With respect to independent claim 1, Appellant argues, *inter alia*, that WebWriter II teaches a document generator that runs inside a web browser, whereas the claimed invention requires a document generator that runs outside (external to) the browser (Reply Br. 15).

With respect to independent claim 22, Appellant argues that neither WebWriter II nor Faustini teaches or suggests the execution of components by the document generator (Reply Br. 22).

With respect to independent claim 26, Appellant argues, *inter alia*, that the combination of WebWriter and Faustini does not teach or suggest generating documents for display by a browser from dynamic documents (Reply Br. 25).

With respect to independent claim 59, Appellant argues, *inter alia*, that the combination of WebWriter and Faustini fails to teach or suggest dynamically editing dynamic web documents, or an editor program having first instructions for requesting that the document generator program processes a dynamic web document (Reply Br. 28-29).

Appellant's contentions, and the Examiner's findings, present us with the following issues:

1. Does the combination of WebWriter II and Faustini teach or fairly suggest a document generator program running at least one of the applications (that run on a data network) being edited and generating the generated documents, as recited in claim 1?

2. Does the combination of WebWriter II and Faustini teach or fairly suggest an editor program dynamically operated on the generated documents displayed by the browser program, as recited in claim 1?

3. Does the combination of WebWriter II and Faustini teach or fairly suggest a document generator program having instructions for executing components on document templates, as recited in claim 22?

4. Does the combination of WebWriter and Faustini teach or fairly suggest a server computer running an application to modify dynamic documents on the server computer, as recited in claim 26?

5. Does the combination of WebWriter and Faustini teach or fairly suggest an editor program for dynamically editing dynamic web documents, as recited in claim 59?

6. Does the combination of WebWriter and Faustini teach or fairly suggest an editor program comprising first instructions for requesting that the document generator program processes a dynamic web document during editing, as recited in claim 59?

FINDINGS OF FACT

Appellant's Specification

1. Appellant's invention creates server side internet applications by placing interactive server side components (ISSCs) on internet pages (Spec. 5).

2. The ISSC algorithm can be extended to create a browser based editor that can place ISSCs on pages and to modify their properties: During dynamic page generation, scripts and handles are embedded into the page that permit editing of the page itself (*id.*).

Faustini

3. Faustini teaches a programming environment and appropriate tools therefor that include the capability for dynamic editing of component objects (col. 5, ll. 9-13).

4. Faustini teaches that its invention provides the dynamic editing capability as soon as the component is instantiated or a component editing request is made (col. 5, ll. 15-17).

5. Faustini teaches the dynamic editing of Java applets and applications (col. 9, ll. 9-10; col. 10, ll. 13-26).

PRINCIPLES OF LAW

> Section 103(a) forbids issuance of a patent when "the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains."

KSR Int'l Co. v. Teleflex Inc., 550 U.S. 398, 406 (2007). The question of obviousness is resolved on the basis of underlying factual determinations including (1) the scope and content of the prior art, (2) any differences between the claimed subject matter and the prior art, (3) the level of skill in the art, and (4) where in evidence, so-called secondary considerations. *Graham v. John Deere Co.*, 383 U.S. 1, 17-18 (1966). *See also KSR*, 550 U.S. at 407 ("While the sequence of these questions might be reordered in

any particular case, the [*Graham*] factors continue to define the inquiry that controls.").

ANALYSIS

CLAIMS 1, 2, 22, 23, 41, AND 42

The Examiner finds that WebWriter II teaches all the features of the invention recited in independent claims 1 and 22, except for editing running applications for generated documents (Ans. 26). The Examiner finds that Faustini supplies this missing teaching (*id.*).

We disagree with the Examiner's conclusion that the combination of WebWriter II and Faustini renders the claimed invention obvious. We agree with Appellant that the disclosed and claimed invention is directed to the editing of *server*-side dynamic web applications that generate web pages for transmission to and display by a web browser (Reply Br. 3; FF 1, 2). Appellant correctly characterizes Faustini as being directed to the editing of java applets and java components, which are *client*-side components (i.e., components run at the client) (FF 5). Accordingly, Faustini's teaching of "dynamic editing capability as soon as the component is instantiated or a component editing request is made" (FF 4) has to do with the editing of a running client-side component. We find that Faustini is not directed to editing software applications that run on a data network and where, upon request by a browser program, at least one of the applications generates documents for display by the browser program, as claim 1 requires.

With respect to independent claim 22, we further agree with Appellant that neither WebWriter II nor Faustini teaches a document generator program for processing document templates and executing components.

We therefore conclude that the Examiner erred in rejecting claims 1, 2, 22, 23, 41, and 42 as being unpatentable over WebWriter II in view of Faustini. We will not sustain the rejection.

CLAIMS 26, 30, 32, 33, 59-63, 67-69, AND 71-73

For reasons similar to those expressed *supra* with respect to claims 1 and 22, we do not agree with the Examiner's conclusion that the combination of WebWriter and Faustini renders obvious the invention recited in independent claims 26 and 59. Because we agree with Appellant that Faustini teaches an editor for client-side components, we find that neither WebWriter nor Faustini discloses a server computer running an application to modify dynamic documents on the server computer, as claim 26 requires. Similarly, we find that neither WebWriter nor Faustini discloses an editor program for dynamically editing dynamic web documents, said documents operating by being transformed into an end user's view upon request by a web browser, as claim 59 requires.

We therefore conclude that the Examiner erred in rejecting claims 26, 30, 32, 33, 59-63, 67-69, and 71-73 as being unpatentable over WebWriter in view of Faustini. We will not sustain the rejection.

CONCLUSIONS

1. The combination of WebWriter II and Faustini does not teach or fairly suggest a document generator program running at least one of the applications (that run on a data network) being edited and generating the generated documents, as recited in claim 1.

2. The combination of WebWriter II and Faustini does not teach or fairly suggest an editor program dynamically operated on the generated documents displayed by the browser program, as recited in claim 1.

3. The combination of WebWriter II and Faustini does not teach or fairly suggest a document generator program having instructions for executing components on document templates, as recited in claim 22.

4. The combination of WebWriter and Faustini does not teach or fairly suggest a server computer running an application to modify dynamic documents on the server computer, as recited in claim 26.

5. The combination of WebWriter and Faustini does not teach or fairly suggest an editor program for dynamically editing dynamic web documents, as recited in claim 59.

6. The combination of WebWriter and Faustini does not teach or fairly suggest an editor program comprising first instructions for requesting that the document generator program processes a dynamic web document during editing, as recited in claim 59.

ORDER

The Examiner's rejection of claims 1, 2, 22, 23, 26, 30, 32, 33, 41, 42, 59-63, 67-69, and 71-73 is reversed.

REVERSED

babc